FIRST EXPERIENCE ON BLUE GENE/L

Stéphane Ethier

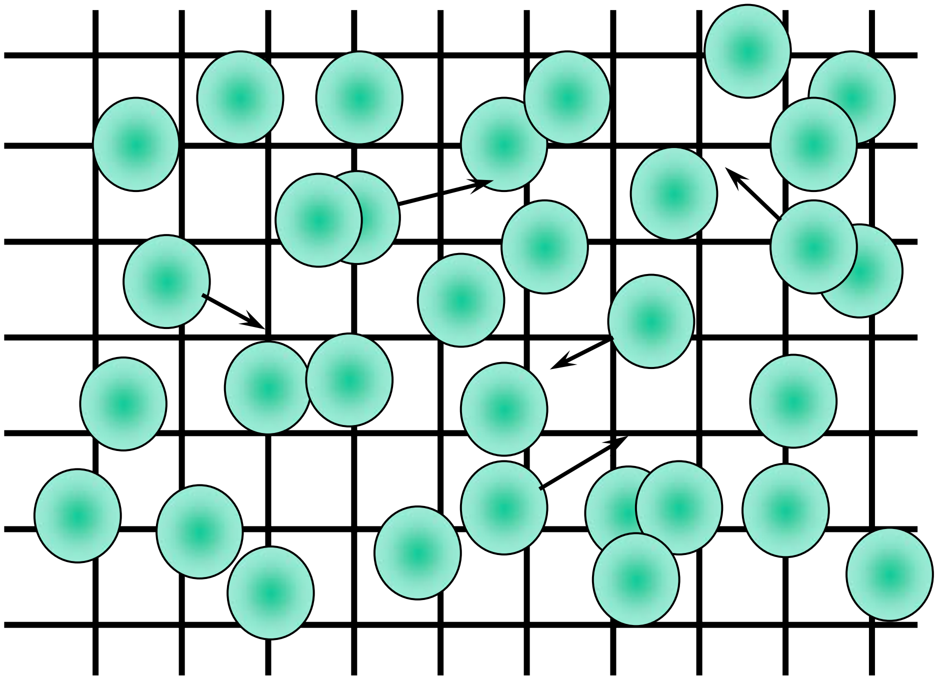
Princeton Plasma Physics Laboratory

Blue Gene Applications Workshop
Argonne National Laboratory
April 27-28, 2005

Thanks to ANL MCS Andrew Siegel for helping me port and run GTC on Blue Gene!!!

GTC: a 3D gyrokinetic particle-in-cell (PIC) code in toroidal geometry

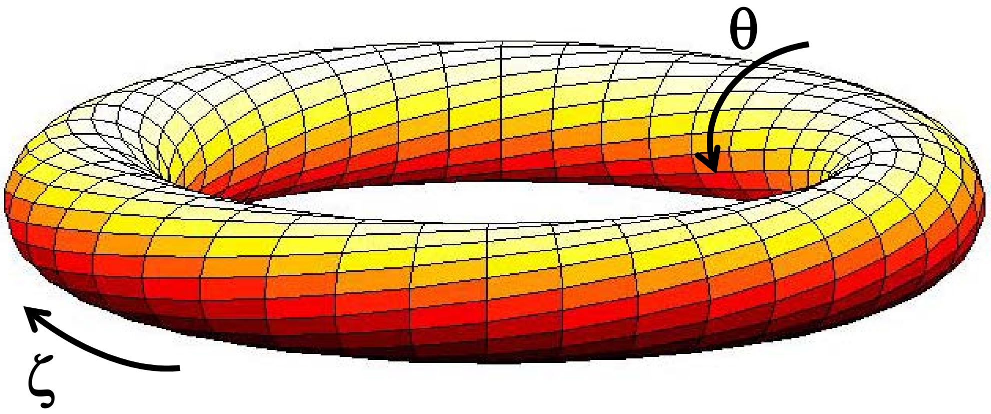
- Particles sample distribution function (markers).
- The particles interact via a grid, on which the potential is calculated from deposited charges.

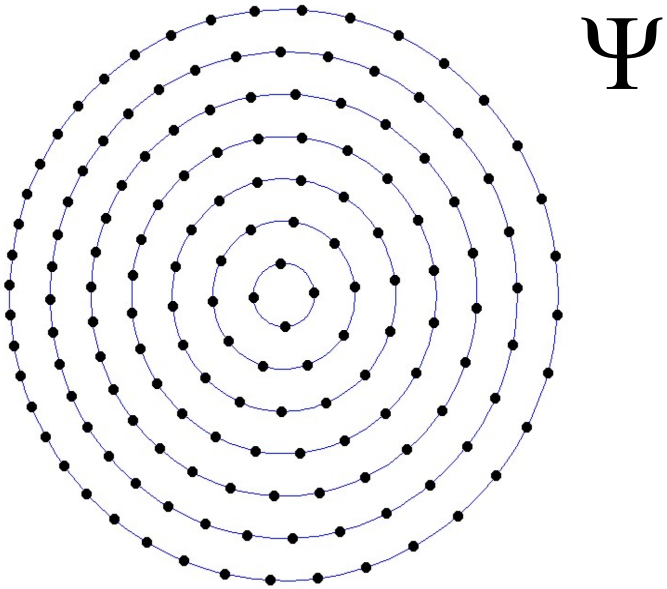


<u>The PIC Steps</u>

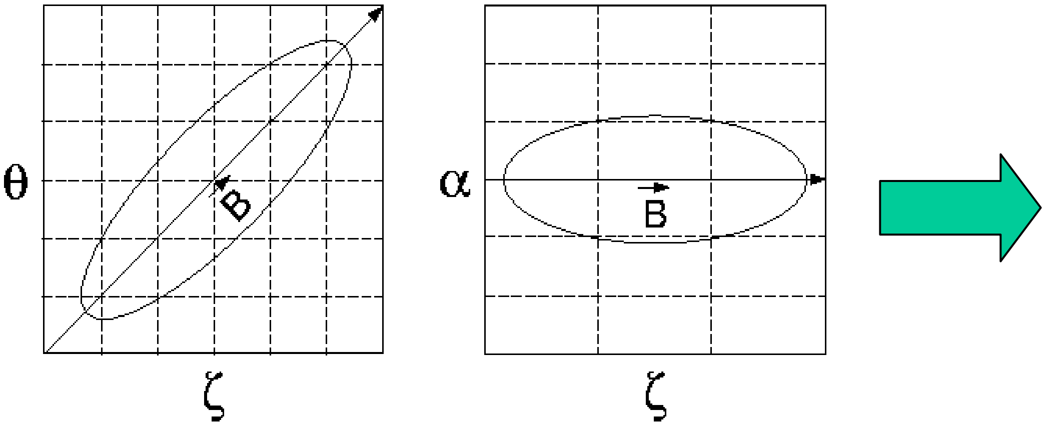
- **"SCATTER", or deposit, charges on the grid (nearest neighbors)**
- **Solve Poisson equation**
- **"GATHER" forces on each particle from potential**
- **Move particles (PUSH)**
- **Repeat…**

GTC mesh and geometry: Field-line following coordinates





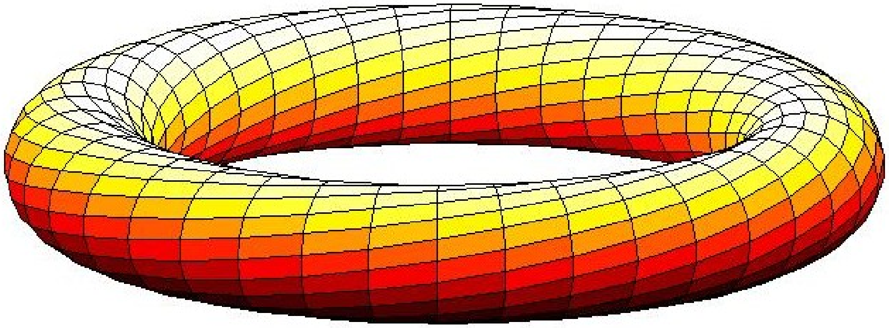
$(\Psi,\alpha,\zeta) \Rightarrow \alpha = \theta - \zeta/q$



Saves a factor of about 100 in CPU time

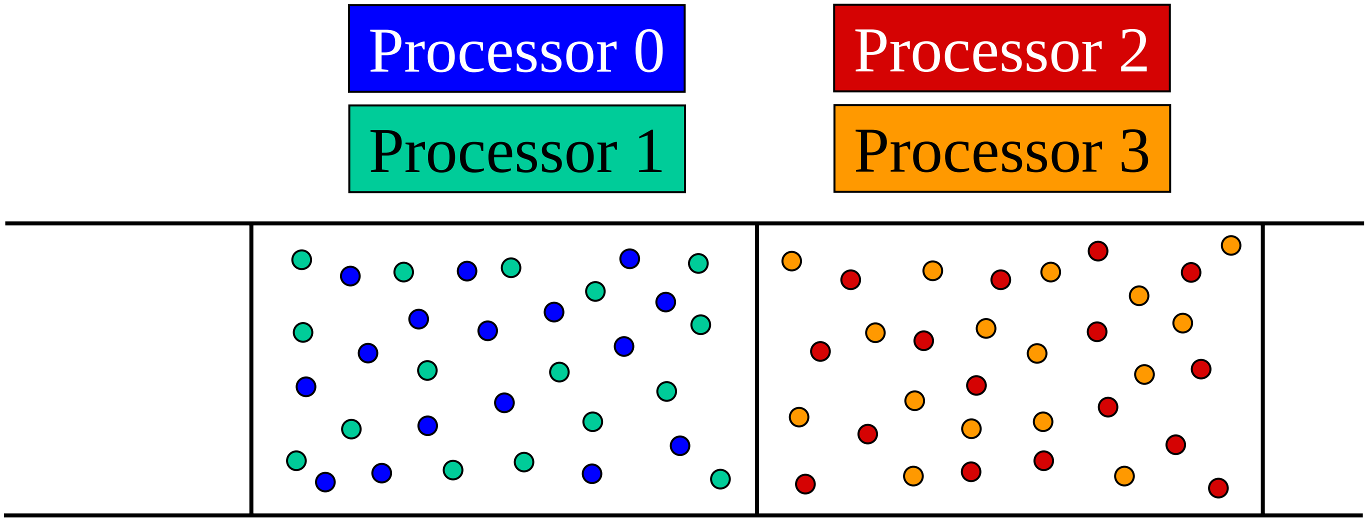
Domain Decomposition

- Domain decomposition:
  - each MPI process holds a toroidal section
  - each particle is assigned to a processor according to its position
- Initial memory allocation is done locally on each processor to maximize efficiency
- Communication between domains is nearest neighbors (MPI_Sendrecv calls).



MPI-based particle decomposition

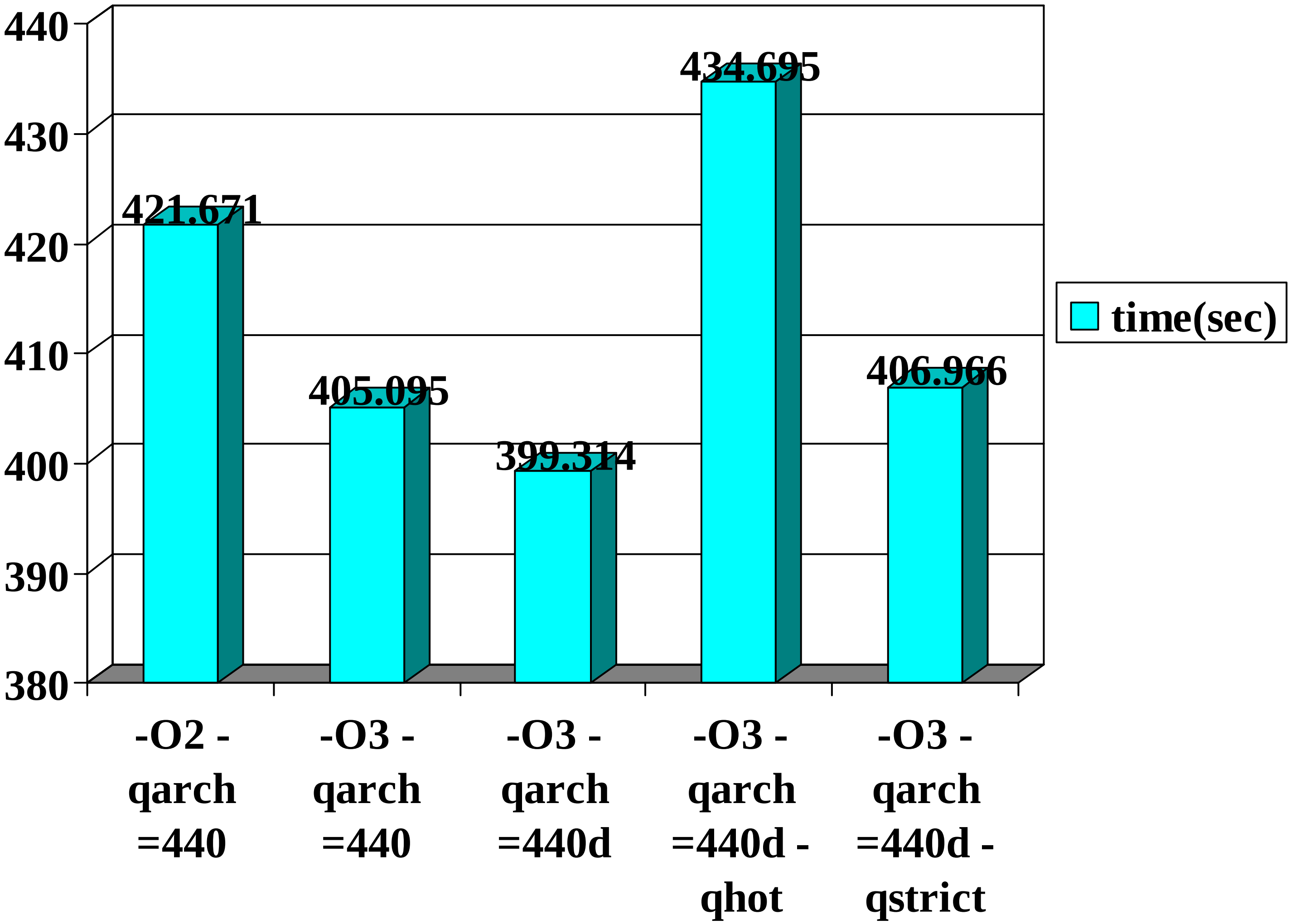
- Each domain in the 1D (and soon 2D) domain decomposition can have more than 1 processor associated with it.
- Each processor holds a fraction of the total number of particles in that domain.



Compilation on Blue Gene/L

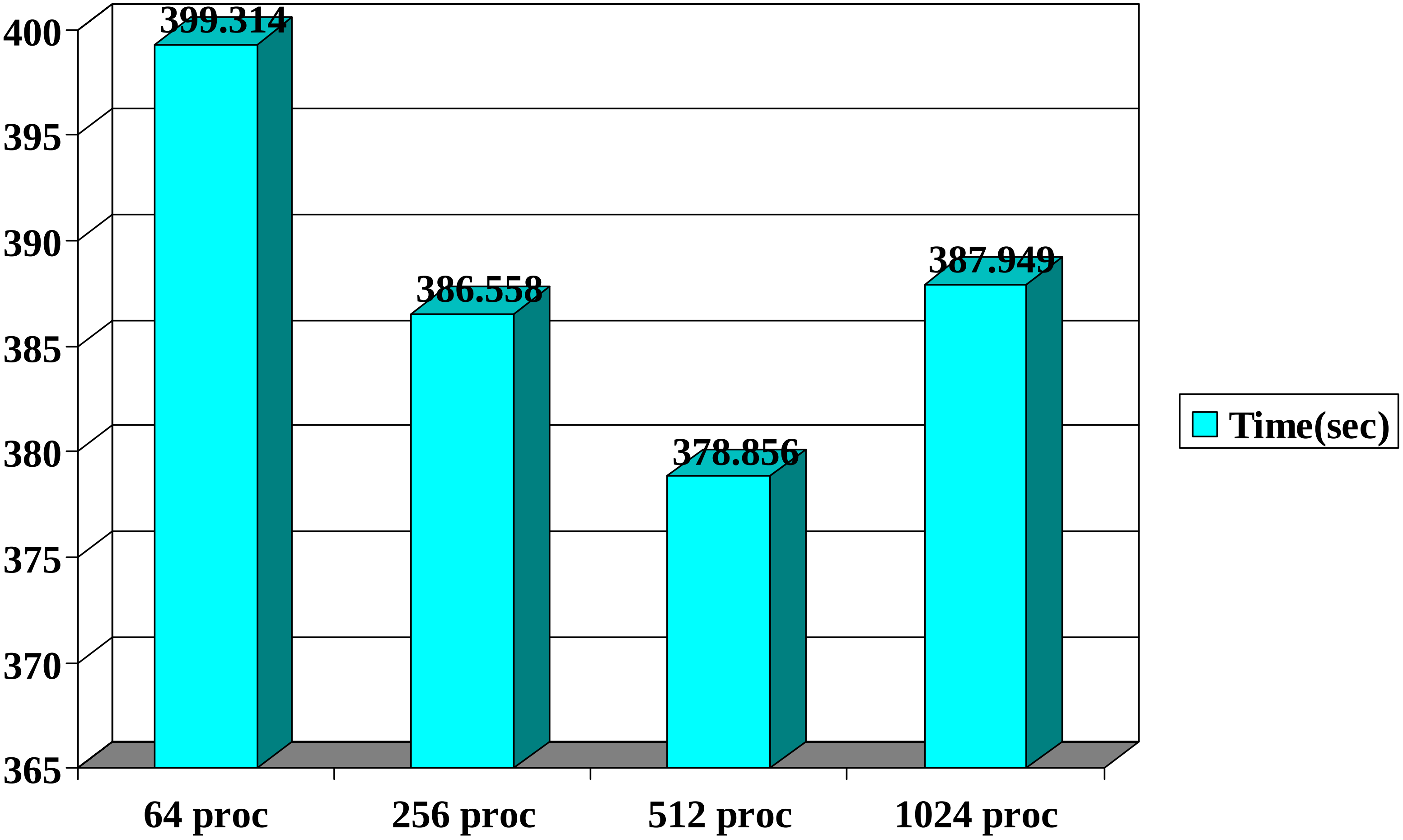
- Very easy… GTC is written in standard Fortran 90/95 and is being run routinely on NERSC’s IBM SP P3.
- Only one issue
  - No MPI Fortran 90 module (cannot do “use mpi”)
  - Had to replace by “ include ‘mpif.h’ “
- Best results with “ –O3 –qarch=440d”
- “-qhot” was the slowest

Compiler flags study
(tested with 64 processor run)



Weak scaling test: same number of particles per processor (~324,000)

64-processor test on Seaborg IBM P3 takes 390.017 sec



Things to know about MPI TRACE

- When running with MPI TRACE (libmpitrace.a), the code produces a maximum of 4 files:
  - mpi_profile.0 for rank 0 process
  - mpi_profile.# for the process with the MINIMUM comm.
  - mpi_profile.# for the process with the MAXIMUM comm.
  - mpi_profile.# for the process with the AVERAGE comm.
- This way, you get a maximum of 4 files even when running with 1,024 processors.
- I got this information from Bob Walkup…

PPPL PRINCETON PLASMA PHYSICS LABORATORY

Next steps

- Link with libmass.a and see if there is improvement
- Do a more thorough study of the communication timings.
- Learn about the mapping and see if I can take advantage of it.
- Do a full profiling of the code to compare with Seaborg or other platforms.
- Get access to a larger number of processors???